Problem 1. Use structure induction for each case.
* Proof for $\bigstar = \odot$.
(1) (Base) $e = 0$: Obviously satisfy since $n = 0$, $0 \rightsquigarrow \odot$ and $[\![\odot]\!] = \{0\}$.
(2) $e = -e'$:
  (2.1) Let $e \Rightarrow n$ and $e \rightsquigarrow \odot$. Then, $e' \Rightarrow -n$ and $e' \rightsquigarrow \odot$.
  (2.2) By induction hypothesis, $-n \in \{0\}$ so $n = 0$.
(3) $e = \text{succ } e'$, $e = \text{pred } e'$: Obviously satisfy since $e \rightsquigarrow \odot$ is always false.

* Prove for $\bigstar = \oplus$.
(1) (Base) $e = 0$: Obviously satisfy since $0 \rightsquigarrow \oplus$ is false.
(2) $e = \text{succ } e'$:
  (2.1) Let $e \Rightarrow n$ and $e \rightsquigarrow \oplus$. Then, $e' \Rightarrow n - 1$ and $e' \rightsquigarrow \odot$ or $e' \rightsquigarrow \oplus$
  (2.2) Case $e' \rightsquigarrow \odot$: by $\bigstar = \odot$ proof, $n - 1 = 0$ so $n = 1 \in \mathbb{N}$.
  (2.3) Case $e' \rightsquigarrow \oplus$: by induction hypothesis, $n - 1 \in \mathbb{N}$ so $n \in \mathbb{N}$.
(3) $e = -e'$, $e = \text{pred } e'$: Obviously satisfy since $e \rightsquigarrow \oplus$ is always false.

* Prove for $\bigstar = \top$.
(1) (Base) $e = 0$: Obviously satisfy since $0 \rightsquigarrow \top$ is false.
(2) $e = -e'$:
  (2.1) Let $e \Rightarrow n$ and $e \rightsquigarrow \top$. Then, $e' \Rightarrow -n$ and $e' \rightsquigarrow \oplus$ or $e' \rightsquigarrow \top$.
  (2.2) Case $e' \rightsquigarrow \oplus$: by $\bigstar = \oplus$ proof, $-n \in \mathbb{N}$ so $n \in \mathbb{Z}$.
  (2.3) Case $e' \rightsquigarrow \top$: by induction hypothesis, $-n \in \mathbb{Z}$ so $n \in \mathbb{Z}$.
(3) $e = \text{succ } e'$:
  (2.1) Let $e \Rightarrow n$ and $e \rightsquigarrow \top$. Then, $e' \Rightarrow n - 1$ and $e' \rightsquigarrow \top$.
  (2.2) By induction hypothesis, $n - 1 \in \mathbb{Z}$ so $n \in \mathbb{Z}$.
(4) $e = \text{pred } e'$:
  (2.1) Let $e \Rightarrow n$ and $e \rightsquigarrow \oplus$. Then, $e' \Rightarrow n + 1$ and $e' \rightsquigarrow \odot$ or $e' \rightsquigarrow \oplus$ or $e' \rightsquigarrow \top$
  (2.2) Case $e' \rightsquigarrow \odot$: by $\bigstar = \odot$ proof, $n + 1 = 0$ so $n = -1 \in \mathbb{Z}$.
  (2.3) Case $e' \rightsquigarrow \oplus$: by $\bigstar = \oplus$ proof, $n + 1 \in \mathbb{N}$ so $n \in \mathbb{Z}$.
  (2.4) Case $e' \rightsquigarrow \top$: by induction hypothesis, $n + 1 \in \mathbb{Z}$ so $n \in \mathbb{Z}$.

Problem 2. Introduce some claims first, and then prove the claims.
(0) Note that this problem is WRONG. Assume that $n \geq 0$.
(1) Let $A = \{X = m \wedge Y = n \wedge Z = 1\}$, $B = \{Z = m^n\}$, $b = \neg(Y = 0)$, $I = (Z \times X^Y = m^n)$.
(2) Also, let $c = \text{while } \neg(Y = 0) \text{ do } c'$ where $c' = (Z := Z \times X; Y := Y - 1)$.
(3) $\dfrac{\models (A \Rightarrow I)\ \{I\}\ c\ \{I \wedge \neg b\}\ \ \models ((I \wedge \neg b) \Rightarrow B)}{\{A\}\ c\ \{B\}}$
  (α) Claim $\models (A \Rightarrow I)$.
  (β) Claim $\models ((I \wedge \neg b) \Rightarrow B)$.
(4) $\dfrac{\{I \wedge b\}\ c'\ \{I\}}{\{I\}\ \text{while } b \text{ do } c'\ \{I \wedge \neg b\}}$
  (γ) Claim $\{I \wedge b\}\ c'\ \{I\}$.
(δ) Claim the while statement terminates.
(5) $\therefore$ For $m \in \mathbb{Z}$ and $n \geq 0$, $\{A\}\ c\ \{B\}$.

* Proof for the claim α
(1) $A = \{X = m \wedge Y = n \wedge Z = 1\} \Rightarrow (Z \times X^Y = 1 \times m^n = m^n) = I$.

* Proof for the claim β
(1) $(I \wedge \neg b) = ((Z \times X^Y = m^n) \wedge (Y = 0)) \Rightarrow (Z \times X^0 = m^n) \Rightarrow (Z = m^n) = B$.

* Proof for the claim γ
(1) $\{I \wedge \neg b\}\ Z := Z \times X;\ Y := Y - 1\ \{I\}$.
(2) $\{I[(Y-1)/Y]\}\ Y := Y - 1\ \{I\}$.(by assignment)
(3) $\{I[(Y-1)/Y][(Z \times X)/Z]\}\ Z := Z \times X\ \{I\}$. (by sequencing and assignment)
(4) $I[(Y-1)/Y][(Z \times X)/Z] \wedge (\neg Y = 0)$
  $\Rightarrow I[(Y-1)/Y][(Z \times X)/Z] \wedge (Y > 0)$
  $\Rightarrow (Z \times X^{(Y-1)} = m^n)[(Z \times X)/Z] \wedge (Y \geq 0)$
  $\Rightarrow (Z \times X \times X^{(Y-1)} = m^n)$
  $\Rightarrow (Z \times X^Y = m^n) = I$
(5) $\therefore \{I \wedge Y > 0\}\ c'\ \{I\}$.

* Proof for the claim δ
(1) Y will decrease exactly 1 after each c' is executed.
(2) Since $n \geq 0$, Y goes to 0 and then terminate.

Problem 3.a. Define each semantics of E and C.
* For the language E,

$$\frac{}{\langle n,\sigma\rangle \to n},\ \frac{}{\langle x,\sigma\rangle \to \sigma(x)},\ \frac{\langle e_1,\sigma\rangle \to v_1\ \langle e_2,\sigma\rangle \to v_2}{\langle e_1+e_2,\sigma\rangle \to v_1+v_2},$$

$$\frac{\langle e_1,\sigma\rangle \to v_1\ \langle e_2,\sigma\rangle \to v_2}{\langle e_1-e_2,\sigma\rangle \to v_1-v_2},\ \frac{\langle e_1,\sigma\rangle \to v'\ \langle e_2,\sigma[v'/x]\rangle \to v}{\langle \text{let } x=e_1 \text{ in } e_2,\sigma\rangle \to v}$$

* For the command C,
$\langle n_1.n_2.S,\ E,\ \text{add}.C\rangle \to \langle n_1+n_2.S,\ E,\ C\rangle$, $\langle n_1.n_2.S,\ E,\ \text{sub}.C\rangle \to \langle n_1-n_2.S,\ E,\ C\rangle$
$\langle n.S,\ E,\ \text{bind(x)}.C\rangle \to \langle S,\ E[\![n/x]\!],\ C\rangle$, $\langle S,\ E,\ \text{unbind(x)}.C\rangle \to \langle S,\ E \mathbin{/\!\!/} [\![x]\!],\ C\rangle$
$\langle S,\ E,\ \text{push}(x).C\rangle \to \langle E(\!|x|\!).S,\ E,\ C\rangle$, $\langle S,\ E,\ \text{push}(n).C\rangle \to \langle n.S,\ E,\ C\rangle$

* E is an environment that stores ordered list of mapping from a variable to a value, where
(1) $E[\![n/x]\!]$ = store a mapping $(x \to n)$ at the beginning
(2) $E(\!|x|\!) = v$, where first mapping from x to some value v
(3) $E \mathbin{/\!\!/} x$ = remove first mapping from x to some value

Problem 3.b. Compilation rules from E to C.
$n \rhd \text{push}(n)$, $x \rhd \text{push}(x)$

$$\frac{e_1 \rhd C_1\ e_2 \rhd C_2}{e_1+e_2 \rhd C_1.C_2.\text{add}},\ \frac{e_1 \rhd C_1\ e_2 \rhd C_2}{e_1-e_2 \rhd C_2.C_1.\text{sub}},\ \frac{e_1 \rhd C_1\ e_2 \rhd C_2}{\text{let } x=e_1 \text{ in } e_2 \rhd C_1.\text{bind}(x).C_2.\text{unbind}(x)}$$

Problem 3.c. $\forall e \in E,\ \langle e,\sigma\rangle \to v \Leftrightarrow \langle \epsilon,\sigma,C\rangle \xrightarrow{*} \langle v,\sigma,\epsilon\rangle$ where $e \rhd C$

Problem 3.d. Prove for each direction.
* Proof of ($\Rightarrow$)
(1) Use structure induction and proof for $e :=$ (let $x = e_1$ in e_2) only.
(2) Let $e_1 \rhd C_1$, $e_2 \rhd C_2$, $\langle e_1,\sigma\rangle \to v'$ and $\langle e_2,\sigma[v'/x]\rangle \to v$
(3) $\langle \epsilon,\sigma,C\rangle = \langle \epsilon,\sigma,C_1.\text{bind}(x).C_2.\text{unbind}(x)\rangle$ (by compile rule)
$\xrightarrow{*} \langle v',\sigma,\text{bind}(x).C_2.\text{unbind}(x)\rangle$ (by induction hypothesis of C_1)
$\to \langle \epsilon,\sigma[\![v'/x]\!],C_2.\text{unbind}(x)\rangle$ (by semantic of bind)
$\xrightarrow{*} \langle v,\sigma[\![v'/x]\!],\text{unbind}(x)\rangle$ (by induction hypothesis of C_2)
$\to \langle v,\sigma,\epsilon\rangle$ (by semantics of unbind)

* Proof of ($\Leftarrow$)
(1) Use induction on k where $\langle \epsilon,\sigma,C\rangle \xrightarrow{k} \langle v,\sigma,\epsilon\rangle$ and proof for $e :=$ (let $x = e_1$ in e_2) only.
(2) Let $e_1 \rhd C_1$, $e_2 \rhd C_2$
(3) Let $\langle \epsilon,\sigma,C\rangle = \langle \epsilon,\sigma,C_1.\text{bind}(x).C_2.\text{unbind}(x)\rangle \xrightarrow{k'} \langle v',\sigma,\text{bind}(x).C_2.\text{unbind}(x)\rangle$
(3.1) Then, $\langle e_1,\sigma\rangle \to v'$ by applying induction hypothesis. ($\because k' < k$)
(4) Also, let $\langle \epsilon,\sigma[\![v'/x]\!],C_2.\text{unbind}(x)\rangle \xrightarrow{k''} \langle v,\sigma[\![v'/x]\!],\text{unbind}(x)\rangle \to \langle v,\sigma,\epsilon\rangle$
(4.1) Then, $\langle e_2,\sigma[v'/x]\rangle \to v$ by applying induction hypothesis. ($\because k'' < k$)
(5) By (3.1), (4.1) and semantic of let, $\langle e,\sigma\rangle \to v$.